IN THE CLAIMS

For the convenience of the Examiner, all pending claims of the present Application are shown below in numerical order whether or not an amendment has been made.

1. **(Previously Presented)** A method for community data caching comprising:

generating a cache community, the cache community having a plurality of cache shares, each cache share associated with one or more locator identifiers;

establishing a primary distribution of the plurality of cache shares using the locator identifiers, the primary distribution indicating a first allocation of the plurality of cache shares among a plurality of clients;

establishing a secondary distribution of the plurality of cache shares using the locator identifiers, the secondary distribution indicating a second allocation of the plurality of cache shares among the plurality of clients to be used in place of the primary distribution in response to a trigger occurrence;

intercepting a request for content at a cache module;

determining a cache share responsible for the request, the cache share being associated with the cache community;

determining whether the content associated with the request is available at the cache share;

retrieving the content associated with the request from the cache share when the content associated with the request is available at the cache share; and

retrieving the content associated with the request from an origin server when the content associated with the request is unavailable at the cache share and storing the content associated with the request retrieved from the origin server at the cache share.

2. **(Canceled)**

3. **(Canceled)**

4. **(Canceled)**

5. **(Canceled)**

6. **(Previously Presented)** The method for community data caching according to Claim 1, wherein the cache shares respectively comprise a plurality of Internet domain names starting with selected letters of the English alphabet.

7. **(Previously Presented)** The method for community data caching according to Claim 1, wherein each cache share respectively comprises a plurality of Internet domain names.

8. **(Previously Presented)** The method for community data caching according to Claim 1, wherein each cache share respectively comprises a set of Internet domain names.

9. **(Previously Presented)** The method for community data caching according to Claim 1, wherein each cache share respectively comprises a plurality of content items associated with requests to be cached at a particular client associated with the cache community.

10. **(Original)** The method for community data caching according to Claim 1 further comprising generating the request at a first client associated with the cache community, the cache community comprising the first client and a plurality of second clients distinct from the first client.

11. **(Original)** The method for community data caching according to Claim 9, wherein determining a cache share responsible for the request comprises:

comparing the request to a location table associated with the cache module, the location table associating each cache share with a cache location, each cache location comprising a selected one of the second clients; and

determining which location is associated with the request in response to the comparison.

12. **(Original)** The method for community data caching according to Claim 1 further comprising collecting statistical information at the cache module, the statistical information being associated with a client associated with the cache module.

13. **(Original)** The method for community data caching according to Claim 1 further comprising determining a resource limit associated with the cache module.

14. **(Original)** The method for community data caching according to Claim 13, wherein the resource limit comprises a percentage of a resource associated with a client associated with the cache module and wherein the resource comprises any item selected from the group consisting essentially of processor time, bandwidth, storage space and memory associated with the client.

15. **(Original)** The method for community data caching according to Claim 1 and further comprising storing content marked as cacheable at the cache module.

16. **(Original)** The method for community data caching according to Claim 1 and further comprising storing content unless the content is marked as non-cacheable at the cache module.

17. **(Original)** The method for community data caching according to Claim 16 and further comprising expiring content stored at the cache module using a content expiration protocol.

18. **(Previously Presented)** A method for community data caching comprising:

storing content unless the content is marked as non-cacheable at a cache module;

intercepting a request for the content at the cache module;

determining a cache share responsible for the request, the cache share being associated with a cache community;

determining whether the content associated with the request is available at the cache share;

retrieving the content associated with the request from the cache share when the content associated with the request is available at the cache share;

retrieving the content associated with the request from an origin server when the content associated with the request is unavailable at the cache share and storing the content associated with the request retrieved from the origin server at the cache share; and

expiring content stored at the cache module using a content expiration protocol, wherein the content expiration protocol comprises the Internet Cache Synchronization Protocol.

19. **(Original)** The method for community data caching according to Claim 1 and further comprising:

determining whether the origin server is unable provide the content associated with the request;

attempting to retrieve the content associated with the request from the origin server until the origin server is able to provide the content associated with the request; and

retrieving the content associated with the request when the server is able to provide the content associated with the request.

20. **(Original)** The method for community data caching according to Claim 19, wherein determining whether the origin server is unable to provide the content comprises determining whether the origin server is busy.

21. **(Original)** The method for community data caching according to Claim 19, wherein attempting to retrieve the content is performed in the background.

22. **(Previously Presented)** A system for community data caching comprising:

a computer readable memory;

an application stored in the computer readable memory and operable to:

generate a cache community, the cache community having a plurality of cache shares, each cache share associated with one or more locator identifiers;

establish a primary distribution of the plurality of cache shares using the locator identifiers, the primary distribution indicating a first allocation of the plurality of cache shares among a plurality of clients;

establish a secondary distribution of the plurality of cache shares using the locator identifiers, the secondary distribution indicating a second allocation of the plurality of cache shares among the plurality of clients to be used in place of the primary distribution in response to a trigger occurrence;

intercept a request for content at a cache module;

determine a cache share responsible for the request, the cache share being associated with the cache community;

determine whether the content associated with the request is available at the cache share;

retrieve the content associated with the request from the cache share when the content associated with the request is available at the cache share; and

retrieve the content associated with the request from an origin server when the content associated with the request is unavailable at the cache share and storing the content associated with the request retrieved from the origin server at the cache share.

23. **(Canceled)**

24. **(Canceled)**

25. **(Canceled)**

26. **(Previously Presented)** The system for community data caching according to Claim 22, wherein each cache share respectively comprises a set of Internet domain names.

27. **(Previously Presented)** The system for community data caching according to Claim 22, wherein each cache share respectively comprises a plurality of content items associated with requests to be cached at a particular client associated with the cache community.

28. **(Original)** The system for community data caching according to Claim 22, wherein the application is further operable to collect statistical information at the cache module, the statistical information being associated with a client associated with the cache module.

29. **(Original)** The system for community data caching according to Claim 22, wherein the application is further operable to generate the request at a first client associated with the cache community, the cache community comprising the first client and a plurality of second clients distinct from the first client.

30. **(Original)** The system for community data caching according to Claim 29, wherein the application, when determining a cache share responsible for the request, is operable to:

compare the request to a location table associated with the cache module, the location table associating each cache share with a cache location, the cache location comprising a selected one of the second clients; and

determine which location is associated with the request in response to the comparison.

31. **(Previously Presented)** The system for community data caching according to Claim 22, wherein the application is further operable to store content marked as cacheable at the cache module.

32. **(Original)** The system for community data caching according to Claim 22, wherein the application is further operable to store content unless the content is marked as non-cacheable at the cache module.

33. **(Original)** The system for community data caching according to Claim 32, wherein the application is further operable to expire content stored at the cache module using a content expiration protocol.

34. **(Previously Presented)** A system for community data caching comprising:

a computer readable memory;

an application stored in the computer readable memory and operable to:

store content unless the content is marked as non-cacheable at a cache module;

intercept a request for content at the cache module;

determine a cache share responsible for the request, the cache share being associated with a cache community;

determine whether the content associated with the request is available at the cache share;

retrieve the content associated with the request from the cache share when the content associated with the request is available at the cache share;

retrieve the content associated with the request from an origin server when the content associated with the request is unavailable at the cache share and storing the content associated with the request retrieved from the origin server at the cache share; and

expire content stored at the cache module using a content expiration protocol, wherein the content expiration protocol comprises the Internet Cache Synchronization Protocol.

35. **(Original)** The system for community data caching according to Claim 22, wherein the application is further operable to:

determine whether the origin server is unable provide the content associated with the request;

attempt to retrieve the content associated with the request from the origin server until the origin server is able to provide the content associated with the request; and

retrieve the content associated with the request when the server is able to provide the content associated with the request.

36. **(Original)** The system for community data caching according to Claim 35, wherein determining whether the origin server is unable to provide the content comprises determining whether the origin server is busy.

37. **(Original)** The system for community data caching according to Claim 35, wherein attempting to retrieve the content is performed in the background.

38. **(Previously Presented)** A method for community data caching comprising:

generating a cache community, the cache community having a plurality of cache shares, each cache share associated with one or more locator identifiers;

establishing a primary distribution of the plurality of cache shares using the locator identifiers, the primary distribution indicating a first allocation of the plurality of cache shares among a plurality of clients;

establishing a secondary distribution of the plurality of cache shares using the locator identifiers, the secondary distribution indicating a second allocation of the plurality of cache shares among the plurality of clients to be used in place of the primary distribution in response to a trigger occurrence;

intercepting a request for content at a cache module, the cache module having an associated resource limit;

determining the resource limit associated with the cache module in response to an incentive;

determining a cache share responsible for the request, the cache share being associated with the cache community;

determining whether the content associated with the request is available at the cache share;

retrieving the content associated with the request from the cache share when the content associated with the request is available at the cache share; and

retrieving the content associated with the request from an origin server when the content associated with the request is unavailable at the cache share and storing the content associated with the request retrieved from the origin server at the cache share.

39. **(Original)** The method for community data caching according to Claim 38, wherein the resource limit comprises a percentage of a resource associated with a client associated with the cache module and wherein the resource comprises any item selected from the group consisting essentially of processor time, bandwidth, storage space and memory associated with the client.

40. **(Original)** The method for community data caching according to Claim 38, wherein the incentive comprises a financial incentive.

41. **(Previously Presented)** A system for community data caching comprising:

a computer readable memory;

an application stored in the computer readable memory and operable to:

generate a cache community, the cache community having a plurality of cache shares, each cache share associated with one or more locator identifiers;

establish a primary distribution of the plurality of cache shares using the locator identifiers, the primary distribution indicating a first allocation of the plurality of cache shares among a plurality of clients;

establish a secondary distribution of the plurality of cache shares using the locator identifiers, the secondary distribution indicating a second allocation of the plurality of cache shares among the plurality of clients to be used in place of the primary distribution in response to a trigger occurrence;

intercept a request for content at a cache module, the cache module having an associated resource limit;

determine the resource limit associated with the cache module in response to an incentive;

determine a cache share responsible for the request, the cache share being associated with the cache community;

determine whether the content associated with the request is available at the cache share;

retrieve the content associated with the request from the cache share when the content associated with the request is available at the cache share; and

retrieve the content associated with the request from an origin server when the content associated with the request is unavailable at the cache share and storing the content associated with the request retrieved from the origin server at the cache share.

42. **(Original)** The system for community data caching according to Claim 41, wherein the resource limit comprises a percentage of a resource associated with a client associated with the cache module and wherein the resource comprises any item selected from the group consisting essentially of processor time, bandwidth, storage space and memory associated with the client.

43. **(Original)** The system for community data caching according to Claim 38, wherein the incentive comprises a financial incentive.

44. **(Previously Presented)** A system for community data caching comprising:

means for generating a cache community, the cache community having a plurality of cache shares, each cache share associated with one or more locator identifiers;

means for establishing a primary distribution of the plurality of cache shares using the locator identifiers, the primary distribution indicating a first allocation of the plurality of cache shares among a plurality of clients;

means for establishing a secondary distribution of the plurality of cache shares using the locator identifiers, the secondary distribution indicating a second allocation of the plurality of cache shares among the plurality of clients to be used in place of the primary distribution in response to a trigger occurrence;

means for intercepting a request for content at a cache module;

means for determining a cache share responsible for the request, the cache share being associated with the cache community;

means for determining whether the content associated with the request is available at the cache share;

means for retrieving the content associated with the request from the cache share when the content associated with the request is available at the cache share; and

means for retrieving the content associated with the request from an origin server when the content associated with the request is unavailable at the cache share and storing the content associated with the request retrieved from the origin server at the cache share.

45. **(Previously Presented)** A system for community data caching comprising:

means for generating a cache community, the cache community having a plurality of cache shares, each cache share associated with one or more locator identifiers;

means for establishing a primary distribution of the plurality of cache shares using the locator identifiers, the primary distribution indicating a first allocation of the plurality of cache shares among a plurality of clients;

means for establishing a secondary distribution of the plurality of cache shares using the locator identifiers, the secondary distribution indicating a second allocation of the plurality of cache shares among the plurality of clients to be used in place of the primary distribution in response to a trigger occurrence;

means for intercepting a request for content at a cache module, the cache module having an associated resource limit;

means for determining the resource limit associated with the cache module in response to an incentive;

means for determining a cache share responsible for the request, the cache share being associated with the cache community;

means for determining whether the content associated with the request is available at the cache share;

means for retrieving the content associated with the request from the cache share when the content associated with the request is available at the cache share; and

means for retrieving the content associated with the request from an origin server when the content associated with the request is unavailable at the cache share and storing the content associated with the request retrieved from the origin server at the cache share.